



Goddard Procedural Requirements (GPR)

DIRECTIVE NO. GPR 8700.5A
EFFECTIVE DATE: January 5, 2005
EXPIRATION DATE: July 19, 2009

APPROVED BY Signature: Original Signed by
NAME: Edward J. Weiler
TITLE: Director

COMPLIANCE IS MANDATORY

Responsible Office: 300 / Office of Systems Safety and Mission Assurance

Title: In-house Development and Maintenance of Software Products

PREFACE

P.1 PURPOSE

This procedure establishes the processes associated with the development and maintenance of software products. This includes planning, design, implementation, verification, validation and review activities.

P.2 APPLICABILITY

This procedure applies to the in-house development and maintenance of all Goddard Space Flight Center (GSFC) software products and processes within scope of the GSFC Quality Management System (QMS).

P.3 AUTHORITY

[NPD 1280.1](#), NASA Management System Policy

[NPD 2820.1](#), NASA Software Policies

P.4 REFERENCES

- a. [NPD 1280.1](#), NASA Management System Policy
- b. [NPD 2210.1](#), External Release of NASA Software
- c. [NPD 8730.4](#), Software Independent Verification and Validation (IV&V) Policy
- d. [NPR 1441.1](#), NASA Records Retention Schedules
- e. [NPR 2210.1](#), External Release of NASA Software
- f. [NPR 2810.1](#), Security of Information Technology
- g. [NPR 7120.5](#), NASA Program and Project Management Processes and Requirements
- h. [NPR 8715.3](#), NASA Safety Manual
- i. NASA-Std-8719.13, NASA Software Safety Manual
- j. [GPR 1310.1](#), Customer Commitments and Review
- k. [GPR 1410.1](#), Directives Management
- l. [GPR 1410.2](#), Configuration Management
- m. [GPR 1440.7](#), Records Control

CHECK THE GSFC DIRECTIVES MANAGEMENT SYSTEM AT
<http://gdms.gsfc.nasa.gov> TO VERIFY THAT THIS IS THE CORRECT VERSION PRIOR TO USE.

- n. [GPR 1710.1](#), Corrective and Preventive Action
- o. [GPR 3410.2](#), Employee Competence and Quality Management System Training
- p. [GPR 4520.2](#), Receiving Inspection and Test
- q. [GPR 5100.1](#), Procurement
- r. [GPR 5330.1](#), Product Processing, Inspection, and Test
- s. [GPR 5340.2](#), Control of Nonconformances
- t. [GPR 6400.1](#), Logistics Support
- u. [GPR 8070.2](#), Identification and Application of Statistical Techniques
- v. [GPR 8700.1](#), Design Planning and Interface Management
- w. [GPR 8700.2](#), Design Development
- x. [GPR 8700.4](#), Integrated Independent Reviews
- y. [GPR 8700.6](#), Engineering Peer Review
- z. [GPR 8730.1](#), Calibration And Metrology
- aa. [GSFC Form 19-21, Template for Software Management Plan](#)

P.5 CANCELLATION

None

P.6 SAFETY

Software safety considerations should be addressed under appropriate reviews (see 1.2.1.2 (SCR), 1.2.2.2 (SRR), 1.2.2.5 (SSR), 1.3.3.6 (PDR), and 1.3.4.3 (CDR) for software development, and the corresponding paragraphs of Section 2 for software maintenance). General safety considerations should be addressed during training; see 1.1.2.1 for software development and 2.1.2.1 for software maintenance, respectively. General safety considerations should also be addressed as part of task-specific training in the Software Management Plan; see 2.3.4 of the appendix. See also NPR 8715.3, Appendix D, section 4.g, for a discussion of software safety analysis. See NASA-Std-8719.13 (which will replace NSS 1740.13), "NASA Software Safety Manual," for further information.

P.7 TRAINING

Training of project personnel to be performed is discussed in 1.1.2 for software development and 2.1.2 for software maintenance, respectively. Customer and user training are discussed in 1.6.2.2 for software development and 2.6.2.2 for software maintenance, respectively. Task-specific training for project personnel, as required by the Quality Management System, should be addressed in the Software Management Plan; see 2.3.4 of the Appendix. See also GPR 3410.2 for more on personnel training.

P.8 RECORDS

Specific records to be maintained, as well as documents/information to be maintained under configuration management, are detailed in the Software Management Plan Guide in the Attachment to this GPR. All software development records are to be turned over to the Project Manager, or designee, at the

DIRECTIVE NO. GPR 8700.5A
EFFECTIVE DATE: January 5, 2005
EXPIRATION DATE: July 19, 2009

Page 3 of 51

termination of the development/maintenance activity for retention/disposal in accordance with [NPR 1441.1](#).

P.9 METRICS

Metrics to be collected during software development and software maintenance are discussed in 1.1.1.3 and 2.1.1.3, respectively. Collection of test metrics is addressed in 4.1.3 of the Appendix.

P.10 DEFINITIONS

- a. Build – A version of a system or component that incorporates a specified subset of the capabilities that the final product will provide.
- b. Configuration Management (CM) – A discipline applying technical and administrative direction and surveillance to: identify and document the functional and physical characteristics of a configuration item; control changes to those characteristics; and record and report change processing and implementation status.
- c. Cross-Cutting Activities – Activities that are performed throughout the life of the project and apply to all phases of a development or maintenance project.
- d. Customer – ‘Customer’ includes both internal Goddard customers and external customers.
- e. Design/ Development – For software, this encompasses all the activities that occur during the design, implementation, and testing of the product prior to validation.
- f. Engineering Peer Review (EPR)- A focused, in-depth technical review that supports the evolving design and development of a product subsystem or lower level of assembly. The purpose of an EPR is to add value and reduce risk through expert knowledge infusion, confirmation of approach, and specific recommendations. An EPR provides a penetrating examination of design, analysis, manufacturing, integration, test and operational details, drawings, processes and data.
- g. Functional Specification – A document that specifies a function that a system or system component must be able to perform.
- h. Glue-ware – Software designed to implement integration of government or commercial off the shelf components into a software system.
- i. Integrated Independent Review (IIR)- One of a series of system level reviews conducted at critical project/product milestones in accordance with GPR 8700.4. IIR's build upon the results of a robust set of EPR's.
- j. Object Code – “The output of a code generator (e.g., compiler)... This code may take a variety of forms: an absolute machine-language program, a relocatable machine-language program, an assembly-language program, or perhaps a program in some other programming language.” (Aho and Ullman, Principles of Compiler Design, p. 518.)

- k. Product Development Lead (PDL) – The manager or leader with overall responsibility for managing the software development activities, including managing the technical and organizational interfaces, and forming and leading the product development team (PDT). For a maintenance project the lead is known as the Product Maintenance Lead (PML).
- l. Product Development Team (PDT) – The team with overall responsibility for the software development activities. This team is equivalent to the QMS Product Design Team. For a maintenance project the team is called a Product Maintenance Team (PMT).
- m. Product Manager (PM)- The individual designated as having management responsibility for a product. A Product Manager may be assigned to any directorate and have a title such as project manager, principal investigator, or RTOP manager.
- n. Project – The set of management, administrative, and technical activities leading to the delivery of a Center product in response to customer requirements and in accordance with Agency and GSFC requirements (see NPR 7120.5).
- o. Software Library – A controlled collection of software and related documentation designed to aid in software development, use, or maintenance. (IEEE Std 610.12-1990)
- p. Software Management Plan – A description of the work to be performed and the resources needed to meet a customer’s requirements. The Software Management Plan includes the *design planning information* and the *process management information*. It may be gathered together as a single document, consist of multiple documents, or be a portion of a more comprehensive document, such as a Project, Product, Implementation, or equivalent Plans.
- q. Release – A particular version of a configuration item that is made available for use outside the development team.
- r. Validation – In design and development, validation is the process of examining the product to determine conformity with the user’s functional requirements. This includes product inspections, functional and operational tests, and environmental simulations.
- s. Verification – In design and development, verification is the process of examining the design to determine conformity with the documented design requirements. This includes reviewing documentation prior to release, performing alternate evaluations to verify the original analysis, and performing physical tests of hardware and operational tests of software.

PROCEDURE

In this document, a requirement is identified by “shall,” a good practice by “should,” permission by “may” or “can,” expectation by “will” and descriptive material by “is”. Best practices from past experience are designated as “recommended” and should be thoughtfully considered for adoption by the PDL.

1. PROCESS FOR SOFTWARE DEVELOPMENT

This process description is intended to be used as a starting point in defining the project's technical approach. The activities are not intended to be strictly time sequenced. This allows them to be used to define activities in a variety of project lifecycles. For example, a project that uses an incremental build lifecycle may use the Implementation and Integration activities many times, while a small project with well-known requirements may use the waterfall lifecycle and execute each applicable activity only once. There may be activities that are unique to the project to mitigate special risks or to handle special requirements. These may be added to meet the project's unique needs.

For administrative or institutional software, the responsible Director of may grant waivers to these requirements, using the process described in GPR 1280.1. Note that any software used in any flight or ground system is not considered institutional.

1.1 Cross-Cutting Activities

1.1.1 Team Management

1.1.1.1 Develop a Software Management Plan

The Product Development Lead (PDL) shall develop a Software Management Plan (SMP) using the Software Management Plan Guide in the Attachment to this GPR. [GSFC Form 19-21](#) provides a template for the recommended SMP outline. (Note: When using [GSFC Form 19-21](#), information can be rearranged or supplemented as long as required information is included.) The completed SMP describes the customer requirements and the resources (e.g., budget, schedule, and staffing) allocated to execute the Plan. The Plan defines the approach that will be used to track project status, for configuration management, product assurance, risk management, and independent verification and validation. It also summarizes the technical approach including the review plan, documentation to be produced and the criteria used to tailor this Process for Software Development. Tailoring is determined by the PDL with the concurrence of line management and the customer (typically a project manager or principal investigator). Tailoring factors include project characteristics such as the criticality of the application, the size of the PDT and user community, the degree of reuse, and other project specific factors

GPR 8700.4 and GPR 8700.6 define the procedures and guidelines for required mission-level reviews and their applicability. The software PDT participates in, or contributes material to, required mission-level Project or Program reviews (e.g. EPR's or IIR's), as required.

The PDL shall define, with the participation of line management, Product Manager, and the customer, an appropriate set of reviews as a resource to increase the probability of success. Reviews may be combined to improve value or efficiency. However, when reviews are combined, review objectives from each shall be addressed to the level of detail required for the individual reviews. The following reviews for software development projects should be considered:

- Software System Concept Review (SCR)
 - Software System Requirements Review (SRR)
 - Software Specification Review (SSR)*
 - Preliminary Design Review (PDR)*
 - Critical Design Review (CDR)
 - Acceptance Test Readiness Review (ATRR)
 - Operational Readiness Review (ORR)*
- * **Required for all projects**

All software PDT reviews, above, shall:

- Be scheduled with published agendas.
- Have a review team chair and other members with appropriate expertise that are independent of the PDT, project and immediate line management
- Include customer representatives
- Record meeting notes
- Collect Requests for Action (RFA's)
- Record and track RFA's to resolution with the independent review team.
- Report results to the Project Manager and the Code 301 Integrated Review (System Review) Team Chair in the case of an activity that is part of a larger system.

1.1.1.2 Manage the Task

The PDL negotiates project-staffing assignments, assigns technical work to staff members, and helps assure that all PDT members are qualified to perform their assigned duties. The PDL is also responsible for the day-to-day management of the task according to the SMP. The PDL shall periodically review, record and report the status of system design, development, implementation, and testing using the methods defined in the SMP. It is recommended that project status be an objective measure of work products accomplished against planned resource allocation (e.g., schedule, cost and effort).

1.1.1.3 Collect Metrics

Project metrics, at a minimum, shall include schedule planned vs. actual dates, budget (effort and cost), product size, and product error information, such as open/closed nonconformances (i.e., discrepancy reports), and shall be collected periodically (e.g., monthly or quarterly) as defined in the SMP. Project metrics are analyzed and the results used to initiate process improvement activities.

1.1.1.4 Document Lessons Learned

The PDL shall query the NASA Lessons Learned Information System (which is maintained at <http://llis.nasa.gov>) and other knowledge resources, as appropriate, to access relevant past experiences and knowledge that can be leveraged to reduce risk, improve quality and efficiency. These queries shall be conducted at the beginning of, and then periodically throughout, the software development lifecycle. The PDL shall also submit significant lessons learned to the web-based LLIS throughout the product lifecycle, as appropriate.

It is recommended that lessons learned be documented at the end of each phase in a Software Development History. The Software Development History is updated to include lists of the products produced, milestones and key events, phase duration, key decisions, problems encountered, and summaries of the metrics collected during the phase. It describes the specific lessons and recommendations that pinpoint the major strengths and weaknesses of the process used and the product itself, with particular attention to planning, requirements, development, testing, CM, QA, and new technology.

1.1.1.5 Prepare and Maintain Project Documentation

The PDT shall design, prepare, and maintain the documents as specified in the SMP and consistent with higher-level Project requirements. The PDT ensures that each document is reviewed, that the changes identified during the review are properly implemented and that appropriate approval signatures are obtained. Well in advance of delivery, the PDT shall complete GSFC Form 1679, "New Technology Report;" this form allows the commercialization office to make an assessment of the software before delivery. It is important not to wait until delivery to complete this form, so that the software may be released in a timely fashion.

1.1.1.6 Transition to Maintenance Organization

At the conclusion of the project the completed system may be transitioned to the internal or external customer, or another organization, for ongoing operations and maintenance. To protect Government-owned intellectual property, this transition should take place in accordance with the provisions of NPD/NPR 2210.1 where applicable (see section 1.6 below). If such a transition occurs, the PDT shall perform training and maintenance activities as defined in the SMP and shall prepare the system and documentation for final delivery. Configuration management, quality assurance and delivery activities are performed as for delivery of a build/release. Project records are to be stored, transferred or disposed of according to the SMP and NPR 1441.1.

1.1.2 Training

1.1.2.1 Identify and Document Required Training Needs

The PDL shall identify and document the QMS-required training needed by PDT members. Required training includes familiarizing PDT members with the SMP, the methodology, standards, and design

CHECK THE GSFC DIRECTIVES MANAGEMENT SYSTEM AT
<http://gdms.gsfc.nasa.gov> TO VERIFY THAT THIS IS THE CORRECT VERSION PRIOR TO USE.

DIRECTIVE NO.	<u>GPR 8700.5A</u>
EFFECTIVE DATE:	<u>January 5, 2005</u>
EXPIRATION DATE:	<u>July 19, 2009</u>

process used, maintenance of PDT records, and use of a nonconformance recording system. It also includes training required when working in proximity to instruments or spacecraft, such as Electrostatic Discharge Awareness Training, Range or Launch Safety, Laser Safety, etc. The PDL works with the appropriate supervisors to assure that the identified training is provided. It is recommended that the PDL identify any special developmental training such as language and tool training, and identify on-the-job-training to build an adequate level of understanding in all PDT members

1.1.2.2 Record Training Received

As PDT members obtain training, the PDL shall maintain records of the training that has been received and by whom (see GPR 3410.2).

1.1.3 Configuration Management (See GPR 1410.2)

1.1.3.1 Perform Configuration Identification

The PDT shall identify, in the SMP, the types of items to be placed under configuration control and shall identify when they will be placed under configuration control. Items that are to be controlled at the higher-level Project shall be identified. The PDT defines baselines for major stages in system development, e.g., requirements baseline, design baseline. The PDT identifies the products comprising each baseline, down to the level of the smallest controllable unit. The PDT shall develop a unique identification for each planned system build/release and delivery. The PDT defines naming and labeling conventions as appropriate. It is recommended that the PDT clearly identify reused software components and that the identification scheme distinguish software components that have been modified from those that are being reused without modification.

1.1.3.2 Maintain Configuration Control

The PDT shall preserve the integrity of all system baselines, components, and products that are not under higher-level Project control. The PDT shall track changes to controlled products to assure that the configuration of all identified items is known at all times. Controlled products include system baselines, project documents (requirements, design, test plans, etc.), source code, object code, released and developmental versions of the target system, critical test software and any customer-provided product (e.g., government off the shelf [GOTS]) used in the development of the software system. The use of a commercial configuration management tool is recommended for configuration control of the project's permanent source code libraries. The PDT shall maintain records of all software configuration management activities performed, including all changes made to software and documents under configuration control.

1.1.3.3 Monitor and Report Configuration Status

It is recommended that the PDT periodically produce configuration status reports. These reports typically include such information as the number of changes made to date, the reason for each change, the number of system releases to date, the functionality provided with each release, and the latest version

CHECK THE GSFC DIRECTIVES MANAGEMENT SYSTEM AT
<http://gdms.gsfc.nasa.gov> TO VERIFY THAT THIS IS THE CORRECT VERSION PRIOR TO USE.

and revision identifiers. It is recommended that the first configuration status report list configuration items for any components that are to be reused. This report is reviewed against design documentation and configuration records from the system that is the source of the software to be reused to ensure that all components are accounted for and of the proper version.

1.1.4 Quality Assurance

1.1.4.1 Support Project Reviews, Walkthroughs, and Inspections

Quality reviews address completeness, readability, traceability, and conformance to PDT standards. They assess both the quality of the product (the deliverable under review) and the quality of the process (the way the review is conducted, including planning, announcement, preparation, conduct, and follow-up). Problem areas are identified and appropriate corrections suggested. Although quality is a shared responsibility, it is recommended that the responsibility for advocating and assessing the quality assurance process be centralized in a single individual. It is recommended that product quality be addressed during all major reviews (requirements reviews, design reviews, etc.). It is recommended that quality also be addressed during in-process reviews (walkthroughs, inspections, test reviews, etc.). It is recommended that inspection criteria for any customer-provided product (e.g., GOTS) used by the PDT be established in conjunction with the customer. Results of all reviews and inspections, including those of incoming customer-supplied products, shall be documented.

1.1.4.2 Review Project Deliverables

It is recommended that all major project deliverables (documents, software releases, presentations, etc.) be reviewed for quality. It is recommended that a sample of minor deliverables (unit designs, unit code, test plans and results, etc.) be reviewed for quality.

1.1.4.3 Monitor and Manage Risk

The Risk Management process is identified in the SMP. Risks shall be continuously identified, analyzed, planned, tracked, controlled, communicated and documented. A detailed discussion of the NASA process and resources for Continuous Risk Management is found in NPR 7120.5 "NASA Program and Project Management Processes and Requirements."

1.1.4.4 Monitor and Support Problem Identification and Corrective Action

The PDT shall employ and follow a problem identification and corrective action process consistent with the Goddard requirements. If a minor Nonconformance Reporting /Corrective Action (NCR/CA) system is used it should include the version or release number where the problem was found and, ideally, the version number that includes the corrections. The PDT shall record the number and severity of defects and nonconformances and the corresponding corrective and preventive actions; this information is used for process improvement. The PDT shall identify and segregate any nonconforming builds/releases, or deliveries. When, with customer concurrence, software is delivered with known nonconformances, the release letter clearly identifies the nonconformances. Software with nonconformances that has not been

DIRECTIVE NO.	<u>GPR 8700.5A</u>
EFFECTIVE DATE:	<u>January 5, 2005</u>
EXPIRATION DATE:	<u>July 19, 2009</u>

approved for delivery is identified and maintained separately so that it is not inadvertently included in a delivered product. The PDT shall document any nonconformances for customer-supplied products (e.g., GOTS). After product delivery major nonconformances which meet the criteria specified in GPR 1710.1 shall be recorded in the in the Center NCR/CA system.

1.1.4.5 Control Documents and Records

The PDT shall control all of the records identified in the SMP. The PDT shall document and implement a process for the identification, review, approval, distribution, retention, and disposition of documents and records.

1.1.4.6 Perform Configuration Audits

Prior to each release or delivery, it is recommended that an audit of the software be performed. The audit verifies that every delivery item (e.g., program, input file, test software, or document) is as reported in the delivery documentation and release letter. Each item on the delivery list is checked to ensure that the item is present, is complete, is the correct version, is in the specified delivery format, and is correctly identified. (Additional requirements apply for external releases or deliveries; refer to 1.6.1 for additional information.)

1.1.4.7 Support IV&V Activities as Required

If it is determined that the project requires IV&V support, the PDT shall provide the required support to the IV&V facility for these activities throughout the life of the project.

1.2 Requirements Activities

1.2.1 Software/System Concept Definition

1.2.1.1 Develop a System, Software and Operations Concept

It is recommended that the PDT develop system, software and operation concepts if they do not already exist. In performing this activity the PDT formulates overall concepts for the system and the software by examining customer needs, looking for similarities in previous missions or systems, and identifying existing software, including commercial off the shelf (COTS) or GOTS that could be used or reused. The system and software concepts take the form of a high-level conceptual architecture. Operations concepts take the form of scenarios that show how users operate the system for each major operational mode.

Where software is to be reused this activity makes use of existing documentation. Where the architecture is reused and a significant portion of the code is planned for reuse, a goal of this activity is to understand the similarities and differences between the system to be reused and the new system. The PDT validates the software concept of the system selected for reuse against customer needs. COTS/GOTS in the existing system are reviewed for continued suitability. Where gaps, unneeded elements, and areas requiring significant modification are identified the PDT looks for additional compatible existing software

CHECK THE GSFC DIRECTIVES MANAGEMENT SYSTEM AT
<http://gdms.gsfc.nasa.gov> TO VERIFY THAT THIS IS THE CORRECT VERSION PRIOR TO USE.

from other sources, including COTS/GOTS that could be used or reused. Effort is concentrated on functionality that is not met by the system to be reused and areas where significant changes are anticipated.

1.2.1.2 Hold a Software System Concept Review

It is recommended that the PDT hold a SCR, or participate in a higher-level Project SCR. The purpose of the SCR is to bring together experts independent of the PDT, project and immediate line management, customers and other interested parties to examine and influence the proposed system and operations concept before detailed requirements are written. If the SCR is held, the PDL shall collect RFA's and shall track RFA responses and resolutions.

SCR Topics:

- Top level requirements (e.g., customer needs, problem description, higher-level Project requirements, anticipated software safety requirements)
- High level, "ideal," architecture, identification of components to be reused
- Identification of major external interfaces
- Operations concepts for major modes of operation

For a system with significant reuse, and where both the PDT and the customers have experience with the system to be reused, it is recommended that the focus of the review be on similarities and differences between the system planned for reuse and the target system. The review helps to determine if the gaps have been properly identified and addressed.

1.2.2 Requirements Definition

1.2.2.1 Define and Document High-Level Requirements

If the customer has not provided high-level functional/operational requirements, the PDT shall define and document the high-level requirements based on the system concept. The PDT works with the customer to define the system and high-level requirements and to derive requirements down to the subsystem (or equivalent) level. The PDT defines internal and external software interfaces, and determines performance and reliability requirements.

1.2.2.2 Conduct a Software System Requirements Review

It is recommended that the PDT hold a SRR or participate in or contribute to a higher-level Project SRR. At the SRR the high-level requirements and specifications are presented (subject to appropriate non-disclosure agreements) to experts independent of the PDT, project and immediate line management, customers, users, and other interested parties (e.g., managers and subject area experts). If the SRR is held, the PDL shall collect RFA's that result from the SRR and track RFA responses and resolutions. The PDT incorporates feedback from the review into the requirements document(s) and places the documents under configuration management. If the high-level requirements have been provided, the SRR

serves as a forum for the PDT to demonstrate its understanding of the requirements and to identify any TBD's.

SRR Topics

- Detailed review of requirements, concentrating on critical requirements, including software safety requirements
- Traceability of requirements to customer source documents
- Performance, interface and derived requirements.
- Identify risks, TBD's and required resolution dates

1.2.2.3 Analyze Requirements for Omissions, Contradictions and Clarifications

The PDT shall analyze the high-level requirements. Conflicting, ambiguous and infeasible requirements are identified. The PDT classifies each requirement into one or more categories such as, "mandatory", "needs clarification," "information only," etc. Use of the Automated Requirements Measurement (ARM) Tool developed by the Software Assurance Technology Center is recommended. More information about the tool can be found at <http://satc.gsfc.nasa.gov/tools/arm/index.html>.

1.2.2.4 Derive and Document Detailed Requirements and Specifications

The PDT uses appropriate requirements analysis techniques (e.g., object-oriented or functional decomposition techniques) to analyze the requirements. At this stage, the PDT focuses on what the software needs to do rather than how to do it. Software operations concepts are refined to a more detailed level and reports, displays, etc. are defined. The PDT shall identify and document detailed requirements and specifications by identifying the primary input types and formats, and output products needed to satisfy the requirements. The PDT shall produce a detailed specification of the technical requirements for the software product. The form and media for the specification are at the discretion of the PDL. Good requirements are clear, complete, consistent, testable, and traceable to customer requirements. The PDT shall document software/hardware interfaces and interfaces among subsystems developed by different teams in interface control documents (ICD's) as needed (e.g., telemetry formats).

1.2.2.5 Conduct a Software Specifications Review (a.k.a. Software Requirements Review)

The PDT shall conduct a SSR to present the detailed software requirements, specifications and operations concepts presented (subject to appropriate non-disclosure agreements) to experts independent of the PDT, project and immediate line management, customers, users, and other interested parties. The PDL shall collect the RFA's that result from the SSR and track RFA responses and resolutions. The PDT incorporates feedback from the review into the requirements, interface and related documents.

SSR Topics:

- Detailed software requirements
- Software operations concepts
- Selection of COTS components
- Allocation of requirements to components to be reused and/or COTS or GOTS components
- Hardware that may need to be built
- Traceability of detailed requirements to high-level requirements
- Definition of data in external interfaces
- Definition of data transformations
- Initial allocation of requirements to builds (i.e., build plan)
- Risks and prototype plans or results
- Safety critical issues
- External Dependencies
- Identification of TBD's and required resolution dates

For projects where there is extensive reuse, the SSR concentrates on the new and modified components. Details of requirements for the components to be reused without modification are presented in enough detail to provide context.

1.2.2.6 Obtain Customer Concurrence on Requirements

The PDT shall obtain customer concurrence on the requirements. This is often achieved through a formal review, such as an SSR, and includes resolution of RFA's generated by the review. Alternative means of achieving concurrence include a document review and approval cycle or a series of requirements walkthroughs where comments are recorded and incorporated as appropriate. When customer concurrence is achieved the resulting requirements shall be placed under configuration control.

1.2.3 COTS/GOTS Evaluation and Selection (where applicable)

1.2.3.1 Validate Existing COTS/GOTS Selections

For projects with extensive reuse, it is recommended that COTS or GOTS components be validated to determine if they adequately meet the requirements of the new system. In some cases only upgrades or new versions are needed.

1.2.3.2 Determine the Selection Criteria for COTS/GOTS

The PDT shall identify and document, as a matrix or checklist, the attributes that will be used to evaluate potential COTS/GOTS for use in the system. Appropriate criteria include: the degree to which the COTS/GOTS provides required functionality; the estimated total cost of using and maintaining the product vs. the cost of developing and maintaining the same functionality over the full life cycle; the familiarity of the PDT with the product and the cost of the learning curve; and vendor considerations such as stability and customer support.

1.2.3.3 Identify Potential COTS/GOTS Candidates

It is recommended that the PDT select a small number of COTS/GOTS candidates for in-depth evaluation. Identification is accomplished using vendor documentation, industry review and ratings, and previous experience. (Note that in most cases, open competitive procurement of COTS products is required. See GPR 5100.1 for additional information on procurements.)

1.2.3.4 Obtain Demo Versions and Evaluate the Candidate COTS/GOTS

It is recommended that the PDT obtain evaluation versions of candidate COTS/GOTS. The PDT works with vendors to obtain demo versions of candidate products. If demo versions are not available, the PDT works with the vendor to understand how the product meets the project needs. The PDT uses the selection criteria to evaluate the candidates.

1.2.3.5 Document the Selection Criteria and Evaluation Results

For each product that is evaluated, the PDT shall document the results of the evaluation against the selection criteria. The results of the evaluation and selection are reported at the SSR. The report provides a record that shows the key factors in the evaluation and provides insight into the decision-making.

1.2.3.6 Document the Risks Associated With the Use of the Selected COTS/GOTS

It is recommended that the COTS evaluation identify potential risks (cost, schedule, reliability, etc.) posed by the selected COTS/GOTS components. Acceptable risk thresholds and mitigation strategies are identified.

1.3 Design Activities

1.3.1 Procurement

1.3.1.1 Procure and Install Hardware, Software, and Firmware in the Development Environment

The PDT procures and installs hardware and COTS/GOTS in the development environment. It is recommended that any software that will be reused be obtained and installed in the development environment. Software to be reused shall be placed under configuration management when it is installed in the development environment. These efforts often extend into implementation.

1.3.2 Prototyping

1.3.2.1 Perform Prototyping to Reduce Risks

It is recommended that the PDT develop prototypes to mitigate risks such as requirements uncertainty, new hardware, a new development language or environment, or stringent performance requirements. Prototypes to resolve requirements uncertainty often take the form of functional user interface mockups

and occur in parallel with requirements definition through to implementation. Prototypes involving COTS/GOTS to evaluate their ability to meet functional requirements may occur in parallel with requirements analysis. Prototypes to investigate performance risks or to evaluate new technology often occur in parallel with design activities.

1.3.2.2 Document the Prototyping Effort

If the prototype is large or lengthy or if the risks under investigation are significant it is recommended that planning, development, and management of the prototype be documented in a prototype plan. A prototype plan describes the purpose and goals of the prototype, the resources allocated to the prototype, and the prototyping method and procedures. It is recommended that the results of the prototype be documented in an update to the prototype plan.

1.3.3 Preliminary Design

1.3.3.1 Refine Operational Scenarios

Operations concepts or operational scenarios are received from a higher-level Project or are developed as part of system concept definition. It is recommended that operational scenarios be refined or developed from operations concepts. Additional detail and clarifications are added. Scenarios are decomposed into lower levels of detail. Screen layouts and report formats are defined, as are preliminary interface dialogs for both human-computer interfaces (HCI) and systems interfaces.

1.3.3.2 Perform Performance Modeling

If the system has performance requirements that are anticipated to be difficult to meet, it is recommended that performance sizing/modeling/measurement be performed.

1.3.3.3 Prepare High-level Architecture Diagrams

The purpose of preliminary design is to define the high-level architecture that best satisfies the requirements and specifications. The PDT uses appropriate analysis techniques (e.g., object-oriented techniques or functional decomposition) to analyze the requirements. The PDT evaluates design options, weighing choices according to system priorities, such as performance, usability, reliability, or maintainability. The PDT shall generate high-level diagrams of the selected architecture, including any COTS/GOTS and glue-ware. The PDT pays special attention to interfaces among COTS/GOTS and custom software elements.

For high-reuse projects the PDT starts with the high-level architecture of the system to be reused. If there is not adequate high-level design documentation for the system to be reused it is recommended that the PDT create this documentation.

1.3.3.4 Design and Document High-level Functions and Specifications

The PDT shall design and document the principal, or critical, modules in the system. The PDT selects an appropriate analysis and documentation approach to accomplish this (e.g., use cases and package specifications, or prologs and program design language). The PDT prepares specifications for the principal components in the system. For projects with extensive reuse the PDT starts with high-level functions and specifications of the system to be reused, reviewing them to ensure that the requirements of the new system are adequately met. Modifications are made as required and new functions and specifications are developed to meet new requirements.

1.3.3.5 Verify the High-level Design

The PDT shall verify the high-level design. The PDT shall document the procedures it uses for this purpose. Appropriate verification activities include walkthroughs and/or PDT level inspections of the high-level design documentation (e.g., design diagrams, use cases, package specifications, etc.). Follow up actions shall be documented and tracked to closure.

1.3.3.6 Conduct a Preliminary Design Review

The PDT shall hold a PDR to present the system design (subject to appropriate non-disclosure agreements) to experts independent of the PDT, project and immediate line management, customers, users and other interested parties (e.g., managers and experts). The PDL shall collect RFAs that result from the PDR and track RFA responses and resolutions. The PDT incorporates feedback from the review into the preliminary design documentation.

PDR Topics:

- Design
- Design alternatives and identified "ideal" design (hardware & software)
- Design drivers
- Safety critical issues
- Software hazard reduction features
- Risks and prototype results
- Lessons learned (previous lessons applied and new ones learned)
- Reuse/COTS tradeoffs
- Testing strategy
- Traceability of requirements to design elements
- Size estimates and schedule
- Staffing plans
- Updated build plan
- Issues and TBD's, along with resolution dates

For projects where the architecture of another system is being reused, the PDR concentrates on new functions and modifications to the existing architecture.

1.3.4 Detailed Design

1.3.4.1 Develop and Document the Low-level Design

The purpose of low-level design is to complete the design so that it satisfies all the requirements and can be directly implemented in code. It is recommended that the PDT prepare design documentation to the lowest level of detail of the system to produce “code-to” specifications for each module. The PDT selects an appropriate analysis and documentation approach to accomplish this (e.g., use cases and package specifications, or prologs and program design language). The PDT refines the operations scenarios for the system and completes the design of input and output formats, such as displays, reports, and databases.

For projects where there is extensive reuse it is recommended that the PDT use the same analysis and documentation approach as was used for the system that is being reused. This facilitates understanding of the design of the new system as a whole. If there is not adequate low-level design documentation for the system to be reused it is recommended that the PDT create it to the level needed to understand the new system. It is recommended that code-to specifications for modules to be modified be produced. Code-to specifications for modules to be reused without modification need not be produced.

1.3.4.2 Verify the Low-level Design

It is recommended that the PDT verify the low-level design. If PDT level verification of the low-level design is performed, the PDT shall document the procedures it uses for this purpose. Walkthroughs and/or PDT level inspections of the emerging designs are appropriate verification activities. Follow up actions shall be documented and tracked to resolution. The low-level design of modules to be reused without modification need not be verified.

1.3.4.3 Conduct a Critical Design Review

It is strongly recommended that the PDT hold a CDR and participate in or contribute to a higher-level Project CDR. The PDT presents the completed design (subject to appropriate non-disclosure agreements) to experts independent of the PDT, project and immediate line management, customers, users, and other interested parties (e.g., managers and experts). Participants evaluate the design to ensure that it satisfies requirements and is correct, complete, robust, and testable. If the CDR is held, the PDT shall collect the RFA's that result from the CDR and shall track RFA responses and resolutions. The PDT incorporates feedback from the review into the design documentation.

CDR Topics:

- Final design (hardware & software)
- Final Reuse/COTS decisions
- Interface definitions and ICD status
- Prototype results
- Build/development dependencies

- Final build plans
- Updates to Test strategies
- Updates to requirements and design traceability, including traceability to test plans
- Updates to size estimates and schedule
- Staffing plans
- Issues, including software safety issues, risks and TBD's

For projects with extensive reuse and where the review is held at the PDT level, the CDR concentrates on the new or changed functions and modifications to the existing design.

1.4 Implementation Activities

1.4.1 Code New Modules and Modify Reused Modules

PDT members shall code new modules and/or modify reused modules according to the design, using the coding standards or conventions specified in the SMP. This activity also includes implementing tailoring and configuration of COTS/GOTS components.

1.4.2 Verify New and Revised Modules

The PDT shall verify new and revised modules. Code reading, walkthroughs, inspections and unit tests are appropriate verification activities. Records appropriate to the verification techniques shall be kept. Errors are corrected and the module is certified as having been verified. The PDT shall document the procedures it uses for verification and certification.

1.4.3 Integrate Modules

The purpose of integration is to ensure that modules function correctly together. It is recommended that the PDT plan and document the integration process and integration testing activities. The PDT integrates the modules (COTS/GOTS, glue-ware, and/or new modules), and tests the integrated modules to ensure that they function together correctly. The PDT reports and corrects defects and then delivers the integrated system for build/release testing.

1.4.4 Draft System and User Documentation

It is recommended that the PDT produce a draft of the system description and user's guide. At a minimum, the draft user's guide could consist of an outline of material to be incorporated into the operations manual. The system description document is an "as-built" design document.

1.5 Test Activities

1.5.1 Build Testing

1.5.1.1 Prepare a Build/Release Test Plan

The purpose of build/release testing is to verify that the software provides the functionality required of the build/release and is a correct implementation of the design; the emphasis is on functional and performance requirements. A test team, consisting of PDT members or a team independent of the PDT but supported by the PDT, shall prepare a build/release test plan. The test plan documents the procedures to be followed in testing each build/release to ensure that it satisfies requirements. The test plan includes traceability between tests and requirements. It includes regression tests to ensure that previously tested functions are not adversely affected in each new build.

1.5.1.2 Verify the Build/Release Test Plan

The test team shall perform a team level verification of the build or release test plan. The verification inspection ensures (1) that the test procedures completely test the requirements to be satisfied in the build/release and (2) that they are written to maximize reuse in system and/or acceptance testing.

1.5.1.3 Prepare or Procure Test Software

If required for adequate test, the PDT shall build, or procure, special software to generate data sets or simulate missing hardware or software components. The PDT determines the degree to which test software must follow PDT standards.

1.5.1.4 Verify Critical Test Equipment or Software

If test software or test equipment is critical to ensuring the success and validity of software, the equipment or software shall be verified to ensure that it functions correctly. The PDT shall document procedures and tests used to verify test equipment and test software to ensure that the tests can be easily and efficiently repeated.

1.5.1.5 Conduct Build/Release Tests

The test team executes the tests specified in the plan. The results of build/release tests shall be recorded. The PDT analyzes and corrects discrepancies found in build testing. It is recommended that the test team produce a test report after the completion of testing for the build/release.

1.5.2 System Testing

System testing is designed to verify the functionality of the software product against the requirements and specifications in a more realistic operational environment, for example as a "string test" including multiple systems. The test team shall develop a system test plan to verify the end-to-end functionality

of the system in satisfying the requirements and specifications. The plan includes traceability of test cases to requirements and specifies the environment for the system tests. It is recommended that the tests be executed on the integrated spacecraft/hardware system where this is feasible.

For a project with significant reuse the end-to-end system tests often include system tests defined for the system that is being reused. It is recommended that the end-to-end system tests include regression testing of components that are being reused.

1.5.2.1 Verify the System Test Plan

The test team will verify the system test plan in order to ensure that all requirements are adequately tested and that test data can reasonably be made available.

1.5.2.2 Develop Detailed Test Procedures

It is recommended that detailed, step-by-step test procedures be prepared. The test procedures identify each test step, the data to be used, and expected results from each step. It is recommended that the test team use draft system and user documentation in preparing test procedures.

1.5.2.3 Conduct System-Level or End-to-End Tests

The test team executes the test procedures specified in the system test plan. Test results shall be recorded. The test team issues documentation of product and test problems that are identified. The PDT, according to CM procedures, will correct these identified problems.

1.5.2.4 Verify System and User Documentation

During conduct of the system test, the draft of the users' guide is verified, noting such flaws as missing or unclear instructions, contradictions, etc. The PDT shall analyze and correct any problems encountered in the document and shall complete the documentation.

1.5.2.5 Report System-Level or End-to-End Test Results

The test team shall produce a test report after the completion of system testing. The test report identifies the tests that were executed and their results, includes a summary of nonconformances found during testing, and identifies remaining nonconformances.

1.5.3 Acceptance Testing

1.5.3.1 Conduct an Acceptance Test Readiness Review

It is recommended that an ATRR be held. The acceptance test team, which is usually independent of the PDT, meets with the PDT, project and immediate line management, customers, testers, users, and other interested parties to review the results of the testing completed to date, and to evaluate preparedness for

acceptance testing. If any external individuals (not Civil Servants) participate in this review, appropriate non-disclosure agreements must be signed in advance of the review. Outstanding problems that affect acceptance testing are discussed. If an ATRR is held, the RFA's that result from the ATRR shall be collected and tracked to resolution.

ATRR Topics:

- Status of system readiness
- Summary of test results; tests performed, successful tests, known problems, waivers and issues
- Discussion of support to be provided for Acceptance Testing.
- Acceptance test approach including ground rules, tools and reporting (input provided by the Acceptance Test Team)
- Test schedule (input provided by the Acceptance Test Team)

1.5.3.2 Support Acceptance Testing

It is recommended that members of the PDT be available to support Acceptance Testing as conducted by the customer or an independent test organization. This support consists of consulting activities, analysis of test results, and correction of nonconformances as requested.

1.6 Delivery and Support Activities

1.6.1 Delivery

When software is released or delivered to an external organization, it is important that Government-owned intellectual property be protected. Refer to NPD/NPR 2210.1, "External Release of NASA Software," for further information concerning the domestic or foreign release of software created by or for NASA.

1.6.1.1 Prepare the Release Letter

The PDL shall document the delivery of the accepted system release in a release letter to the customer that details the release identification, release contents, release capabilities, and any remaining nonconformances.

1.6.1.2 Prepare Shipping Records

The PDT shall prepare shipping records that accompany any hardware or other items transported. The records identify each item being delivered, its delivery medium, and its destination.

1.6.1.3 Deliver the Completed System or Release

The PDT shall deliver the software and all documentation using the media and methods identified in the SMP. (See NPD/NPR 2210.1 regarding external releases or deliveries.)

1.6.2 Operations Support

1.6.2.1 Support the Operational Readiness Review

The PDT shall support the ORR held by the Acceptance Test Team, or shall participate in, or contribute to, a higher-level Project ORR. The ORR serves as an opportunity to review the ability of the system to support operations. The status of the remaining nonconformances and the status of system documentation are presented. If an ORR is held, the PDL shall collect the RFA's that result from the ORR and shall track RFA responses and resolutions.

ORR Topics:

- Summary of test results including Acceptance Testing with nonconformances and their impact on operations
- Status of system documentation including user and operations manuals.
- Status of external interface agreements
- Readiness of operational environment for installation
- Operational support and maintenance support plans
- Configuration control procedures

1.6.2.2 Provide Customer Training

The PDT shall provide training in the installation and execution of the software to operations staff (or other users) to the extent specified in the SMP. The PDT shall train maintenance personnel as specified in the SMP.

1.6.2.3 Provide Technical Assistance

3. The PDT shall provide technical support and maintenance during the agreed-upon support period as defined in the SMP. The PDT responds to questions raised by operations staff and other users and assists in identifying and analyzing defects or anomalies in the behavior or performance of the system. Records of maintenance requests shall be kept.

2. PROCESS FOR SOFTWARE MAINTENANCE

This process description is intended to be used as a starting point in defining the project's technical approach. The activities are not intended to be strictly time sequenced. This allows them to be used to define activities in a variety of maintenance situations. There may be activities that are unique to the project to mitigate special risks or to handle special requirements. These may be added to meet the project's unique needs.

2.1 Cross-Cutting Activities

2.1.1 Team Management

2.1.1.1 Develop a Software Management Plan

The Product Maintenance Lead (PML) shall develop a SMP using the Software Management Plan Guide in the Attachment of this GPR. [GSFC Form 19-21](#) provides a template for the recommended SMP outline. (Note: When using [GSFC Form 19-21](#), information can be rearranged or supplemented as long as required information is included.) The completed SMP describes the customer requirements and resources (e.g., budget and staffing) allocated to execute the Plan. It documents release planning and the strategy for prioritizing maintenance requests including criteria for identifying emergency changes. It identifies a nominal maintenance release schedule, reviews for nominal releases and their formality, and internal Product Maintenance Team (PMT) level verification. It identifies exceptional circumstances that may result in fewer or more reviews or more formal reviews (e.g., response to a critical spacecraft failure).

For external releases or deliveries, additional requirements may apply. See NPD/NPR 2210.1, "External Release of NASA Software," for further information on the domestic or foreign release of software created by or for NASA.

The SMP defines the approach that will be used to track project status, for configuration management, product assurance, risk management, and independent verification and validation. It also summarizes the technical approach including the review plan, documentation to be maintained and the criteria used to tailor this Process for Software Maintenance.

Tailoring is determined by the PML in consultation with management and the customer. Tailoring factors include project characteristics such as the criticality of the application, the size of the Product Maintenance Team (PMT) and user community, the degree of reuse, and other project specific factors. Required activities (i.e., those designated as "shall") may not be deleted.

The PML defines the review plan for nominal releases based on the criticality, complexity, and the number of users affected by the maintenance changes. For high priority or emergency patches the PML has the authority to accomplish the goals of a review - to achieve customer concurrence - through other means (e.g., telecon or email), followed by more formal documentation of concurrence after the emergency is resolved. The following reviews should be considered:

- Maintenance Release Planning Review
- Maintenance Release Design Review*
- Acceptance Test Readiness Review
- Operations Readiness Review

*Required Reviews

Software project reviews shall:

- Be scheduled with published agendas
- Include customers or customer representatives
- Record meeting notes
- Collect RFA's
- Record and track RFA's to resolution.

2.1.1.2 Manage the Task

The PML negotiates project-staffing assignments, assigns technical work to staff members, and helps assure that all PMT members are qualified to perform their assigned duties. The PML is also responsible for the day-to-day management of the task according to the SMP. The PML shall periodically review and record the status of maintenance activities using the methods defined in the SMP. It is recommended that project status be an objective measure of work products accomplished against planned resource allocation (e.g., schedule, cost and effort).

2.1.1.3 Collect Metrics

Project metrics, at a minimum, shall include schedule planned vs. actual dates, budget (effort and cost), actual release content vs. planned release content, change in product size, and product error information, such as open/closed nonconformances (i.e., discrepancy reports), and shall be collected periodically (e.g., monthly or quarterly). Project metrics are analyzed and the results used to initiate process improvement activities.

2.1.1.4 Document Lessons Learned

The PML shall review, at the beginning of, and then periodically throughout the maintenance effort, the LLIS (which is maintained at <http://llis.nasa.gov>) for relevant experiences and knowledge that can be leveraged to reduce risk, improve quality, an/or apply Best Practices. The PML shall also submit significant lessons learned to the web-based LLIS system as appropriate.

It is recommended that lessons learned be documented in a Software Development History at each maintenance release. The Software Development History is updated to include lists of the products in the delivery, milestones and key events, release duration, key decisions, problems encountered, and summaries of the metrics collected during the phase. It describes the specific lessons and recommendations that pinpoint the major strengths and weaknesses of the process used and the product itself, with particular attention to planning, requirements, development, testing, CM, QA, and changes in technology.

2.1.1.5 Prepare and Maintain Project Documentation

The PMT shall maintain the project documents as specified in the SMP. The PMT ensures that each change to project documentation is reviewed, and that the changes identified during the review are properly implemented and that appropriate approval signatures are obtained.

2.1.2 Training

2.1.2.1 Identify and Document Required Training Needs

The PML shall identify and document the training for PMT members that are required by the QMS. Required training includes familiarizing PMT members with the SMP, the methodology, standards, and design process used, maintenance of PMT records, and use of a nonconformance recording system. It also includes training required when working in proximity to instruments or spacecraft, such as Electrostatic Discharge Awareness Training, Range or Launch Safety, Laser Safety, etc. The PML works with the appropriate supervisors to assure that the identified training is provided. It is recommended that the PML identify any special developmental training such as language and tool training, and identify on-the-job-training to build an adequate level of understanding in all PMT members of software that will be maintained.

2.1.2.2 Record Training Received

As PMT members obtain training, the PML shall maintain records of the training that has been received and by whom (see GPR 3410.2).

2.1.3 Configuration Management (See GPR 1410.2)

2.1.3.1 Perform Configuration Identification

The PMT shall identify, in the SMP, the types of items to be placed under configuration control and shall identify when they will be placed under configuration control. Items, if any, that are to be controlled at the higher-level Project, shall be identified. For maintenance projects each release constitutes a baseline. The PMT identifies the products comprising each baseline, down to the level of the smallest controllable unit. The PMT shall develop a unique identification for each planned system release/version and delivery. The PMT defines naming and labeling conventions as appropriate. It is recommended that the PMT follow naming and labeling conventions (e.g., release/version, file and module naming conventions) consistent with those established during the system development.

2.1.3.2 Maintain Configuration Control

The PMT shall preserve the integrity of all system baselines, components, and products. The PMT shall track changes to controlled products to assure that the configuration of all identified items is known at all times. Controlled products include system baselines, project documents (requirements, design, test plans, etc.), source code, object code, released and developmental versions of the target system, critical

DIRECTIVE NO.	<u>GPR 8700.5A</u>
EFFECTIVE DATE:	<u>January 5, 2005</u>
EXPIRATION DATE:	<u>July 19, 2009</u>

test software and any customer-provided product (e.g., GOTS) used in the development of the software system. The use of a commercial configuration management tool is recommended for configuration control of the project's permanent source code libraries. This also includes maintaining change control over any critical test software. The PMT shall maintain records of all software configuration management activities performed, including all changes made to software and documents under configuration control.

2.1.3.3 Monitor and Report Configuration Status

It is recommended that the PMT periodically produce configuration status reports. These reports typically include such information as the number of changes made to date, the reason for each change, the number of system releases to date, the functionality provided with each release/version, and the latest version and revision identifiers. It is recommended that the first configuration status report list configuration items for the system at the start of the maintenance project and include identification of any known nonconformances that exist at the time. This report is reviewed against design documentation and configuration records received with the software at the start of the project.

2.1.4 Quality Assurance

2.1.4.1 Support Project Reviews, Walkthroughs, and Inspections

Quality reviews address completeness, readability, traceability, and conformance to PMT standards. They assess both the quality of the product (the deliverable under review) and the quality of the process (the way the review is conducted, including planning, announcement, preparation, conduct, and follow-up). Problem areas are identified and appropriate corrections suggested. Although quality is a shared responsibility, it is recommended that the responsibility for advocating and assessing the quality assurance process be centralized in a single individual. It is recommended that product quality be addressed at each major review. It is recommended that quality also be addressed during in-process reviews (walkthroughs, inspections, test results etc.). It is recommended that inspection criteria for any customer-provided product (e.g., GOTS) used by the PMT be established in conjunction with the customer. Results of all reviews and inspections, including those of incoming customer-supplied products, shall be documented.

2.1.4.2 Review Project Deliverables

It is recommended that all major project deliverables (documents, software releases, presentations, etc.) be reviewed for quality. It is recommended that a sample of minor deliverables (unit designs, unit code, test plans and results, etc.) be reviewed for quality.

2.1.4.3 Monitor and Manage Risk

The Risk Management approach is identified in the SMP. Risks shall be continuously identified, analyzed, monitored, tracked, mitigated and status reported at each review. A detailed discussion of risk

management is found in NPR 7120.5 “NASA Program and Project Management Processes and Requirements.”

2.1.4.4 Monitor and Support Problem Identification and Corrective Action

The PMT shall employ and follow a problem identification and corrective action process consistent with the Goddard Center requirements. If a minor NCR/CA system is used it should include the version or release number where the problem was found and, ideally, the version number that includes the corrections. The PMT shall record the number and severity of defects and nonconformances and the corresponding corrective and preventive actions; this information is used for process improvement. The PMT shall identify and segregate any nonconforming builds/releases or deliveries. When, with customer concurrence, software is delivered with known nonconformances, the release letter clearly identifies the nonconformances. Software with nonconformances that has not been approved for delivery is identified and maintained separately so that it is not inadvertently included in a delivered product. The PMT shall document any nonconformances for customer-supplied products (e.g., GOTS). Major nonconformances that meet the criteria specified in GPR 1710.1 shall be recorded in the in the Center NCR/CA system.

2.1.4.5 Control Documents and Records

The PMT shall control all its own records as identified in the SMP. The PMT shall document and implement a process for the identification, review, approval, distribution, retention, and disposition of documents and records.

2.1.4.6 Perform Configuration Audits

At the beginning of the maintenance project and prior to each release or delivery, it is recommended that an audit of the software be performed. The audit at the start of the project ensures that all the required system elements and documentation are present. Audits performed prior to a release or delivery verify that every delivery item (e.g., program, input file, test software, or document) is as reported in the delivery documentation and delivery letter. Each item on the delivery list is checked to ensure that the item is present, is complete, is the correct version, is in the specified delivery format, and is correctly identified. (Additional requirements apply for external releases or deliveries; refer to 2.6.1 for additional information.)

2.1.4.7 Support IV&V Activities as Required

If it is determined that the project requires IV&V support, the PMT shall provide the required support to the IV&V facility for these activities throughout the life of the project.

2.2 Requirements Activities

2.2.1 Maintenance Request Evaluation

2.2.1.1 Document Each Maintenance Request and Assign a Unique Identifier for Tracking Purposes

Each maintenance request shall be assigned a tracking number and logged. The PML assigns PMT members to evaluate the request. It is recommended that the PML assign lead responsibility for investigating each maintenance request to a specific PMT member.

2.2.1.2 Perform Preliminary Assessment of Resources Required to Implement Each Maintenance Request

It is recommended that a preliminary assessment of the resources required to implement a solution to the maintenance request be performed and documented. It is recommended that the PMT document procedures for performing this assessment in the SMP.

2.2.1.3 Classify the Maintenance Request

The SMP identifies recommended classification categories for maintenance requests, such as severity level and type (e.g., by subsystem, by error type). It is recommended that the maintenance request be classified and that classification metrics be collected.

2.2.1.4 Prioritize the Maintenance Requests

The PML shall periodically meet with the customer to establish and accept or reject recommendations for all new change requests. Notification of the rejected changes shall be sent to the originator along with a brief rationale. Accepted maintenance requests shall be assigned a resolution priority based on the user's priority, the PMT's assessment of the impact on operations, and the effort to implement a solution. It is recommended that the SMP include guidelines and procedures for prioritization, including criteria for designating a nonconformance as serious enough to require following emergency patch procedures.

2.2.2 Maintenance Release Definition

2.2.2.1 Derive New or Modified Requirements from the Maintenance Request

If the request is for minor modifications to existing requirements, the PMT identifies the required changes. If the request is for new functionality or for complex changes, the analysis method specified in the SMP is used to derive detailed requirements to fully define the change. The focus of requirements definition and analysis activities is on what the changed software needs to do rather than how to do it. Modifications that affect operations, for example changes to system parameters, reports, displays, etc. are defined. New reports and displays are identified. Test requirements are also derived. If the change is

driven from needs external to the user community (e.g., a change to an institutional or other interfacing system that requires a corresponding change to the system being maintained), walkthroughs with users and representatives of interfacing systems are recommended. Draft updates to the requirements document are prepared using mark-ups to show where text is to be inserted or deleted to make clear what is being changed.

2.2.2.2 Refine the Estimated Resources (e.g., Staff and Effort) Required to Implement the Requested Maintenance Change

It is recommended that the assigned PMT member reevaluate the preliminary impact assessment to make a detailed estimate of the resources and schedule required to implement the change. The resource estimate includes the effort to generate additional or changed test data and to modify test software as required. It is recommended that the PMT identify dependencies among components and determine if a phased implementation is desirable and/or feasible.

2.2.2.3 Plan Release Contents Using Request Priorities and Resource Estimates

The PML shall work with customers to plan the content of maintenance releases. The PML uses knowledge about the areas of the system that will be affected by a change and about upcoming work, along with the customer's priority input, to identify planned release contents. If a change is identified as critical to continued operations, one or several maintenance requests may be designated as an emergency patch and implemented on an expedited schedule.

2.2.2.4 Conduct a Maintenance Release Planning Review

It is recommended that a Maintenance Release Planning Review be conducted for each Maintenance Release. The purpose of the review is to gain customer concurrence on the release contents, the proposed changes to requirements and the planned release schedule. Review participants include customers, users or their representatives. If the Maintenance Release Planning Review is held, meeting minutes shall be taken and RFA's tracked to closure.

Maintenance Release Planning Review Topics:

- Summary of maintenance requests in the release, including priority and resource estimates
- New and modified requirements
- Operational impacts
- Release schedule

In most cases the review is an informal technical meeting. However, it is recommended that the review be formal for major upgrade releases, where there are multiple user organizations, if the changes are complex or have a significant impact on operations, or if the changes impact interfacing systems.

2.3 Design Activities

2.3.1 Update the Maintenance Request Documentation

For each maintenance change in the planned release, it is recommended that information describing the alternatives considered and the selected solution be added to the maintenance request tracking documentation. This information serves as documentation of the proposed change to the system and provides traceability for changes to software and documentation that result from the maintenance request.

2.3.2 Identify all Affected Elements of the Software Design

Identify all software elements, including COTS/GOTS components and documentation that are affected by the maintenance change(s). If a Requirements Traceability Matrix exists, use it as a guide to help identify the components that implement the impacted requirements.

2.3.3 Modify the Design

Using the design approach and techniques specified in the SMP, design modifications to existing components and design new components. The PMT shall document the design modifications. Where possible, use mark-ups of existing design material to highlight changes. If the changes are too extensive to be shown as markups, generate replacement diagrams and other design material, and make both the new material and the old material available for review.

2.3.4 Identify Tailoring/Configuration Changes (where applicable)

If the maintenance change calls for changes to the tailoring or configuration of a COTS/GOTS component, identify and document those changes. If the maintenance request calls for changes to existing system configuration parameters, identify and document those changes.

2.3.5 Procure and Install Replacement COTS/GOTS Components in the Development Environment (where applicable)

If approved by the customer, initiate procurement activities to obtain the new version(s) of COTS/GOTS components and install the product for testing when it arrives. (Note that in most cases, open competitive procurement of COTS products is required. See GPR 5100.1 for additional information on procurements.)

2.3.6 Verify Design Modifications

It is recommended that a PMT-level review of the resulting modified design be conducted. Follow-up actions shall be documented and tracked to resolution.

2.3.7 Conduct a Maintenance Design Review

For each maintenance release, the PMT shall hold a Maintenance Design Review that includes customers and users. If the release includes changes to a critical system, if the changes are complex, or if there are extensive changes to the system design (e.g., architectural changes or replacement, not upgrade, of COTS/GOTS components), then it is recommended that the review be formal. Meeting minutes shall be taken and RFA's tracked to closure.

Design Review Topics:

- Summary of maintenance requests in the release, including priority and resource estimates
- Summary of requirements changes
- Interface changes
- Summary of operational impacts
- Identification of system architectural changes (if any)
- Description of significant design changes
- Changes to COTS/GOTS
- Release test plan overview
- Updated release schedule

2.4 Implementation Activities

2.4.1 Code New Modules and Modify Existing Modules

PMT members shall code new modules and modify existing modules according to the design, using the coding standards or conventions specified in the Software Development Plan. This activity includes implementing changes to COTS/GOTS tailoring and configuration parameters.

2.4.2 Verify New and Revised Modules

The PMT shall verify new and revised modules. Code reading, walkthroughs, inspections and unit tests are appropriate verification activities. Records appropriate to the verification techniques shall be kept. Errors are corrected and the module is certified as having been verified. The PMT shall document the procedures it uses for verification and certification.

2.4.3 Integrate the Release/Patch

The PMT plans the integration activities, integrates the modules (COTS/GOTS, glue-ware, and/or new modules), and tests the integrated modules to ensure that they function together correctly. The PMT documents the procedures it uses for integration. Problems are identified and corrected in preparation for release/patch testing.

2.4.4 Update System and User Documentation if Required

The PMT produces draft updates to the appropriate documentation (e.g., system description, operations manuals, user's guide and training material). At a minimum, the draft updates are mark-ups to the existing documents.

2.5 Test Activities

2.5.1 Release Testing

2.5.1.1 Develop a Release/Patch Test Plan Based on the Existing Test Plans and Regression Tests

The purpose of release/patch testing is to verify that the software provides the functionality required of the release and is a correct implementation of the design; the emphasis is on functional and performance requirements. A test team, consisting of PMT members or a team independent of the PMT but supported by the PDT, shall develop a release/patch test plan using existing system test documentation as the basis for the new and modified tests. Maintenance release tests are at the level of build and functional scenario tests. They demonstrate that the system functions as designed and that changed and new requirements are satisfied. The plan documents the procedures to be followed in testing the release to ensure that it satisfies the new or changed requirements. When testing fixes for defects it is often useful to follow the scenario that was used to recreate the problem as one of the test cases. For critical systems the release test plan shall include regression tests to ensure that unmodified functions are not adversely affected. The plan defines build test completion criteria. The completed test plan shall be placed under configuration management.

2.5.1.2 Review the Test Plan

It is recommended that the test plan be reviewed at the PMT level. If the change is critical or affects interfacing systems a more formal review that includes customer or user representation is recommended. Review meeting notes shall be documented.

2.5.1.3 Prepare Detailed Test Procedures for New Test Cases

Detailed step-by-step test procedures shall be prepared. The test procedures identify each test step, the data to be used and results expected from each step. The detailed test procedures reference and build on existing system test procedures.

2.5.1.4 Modify Test Data and Software as Needed

Prepare changes to test data. If necessary, modify existing test drivers and/or test data generation software and verify the changes.

2.5.1.5 Conduct Patch/Release Tests

The test team executes the tests specified in the plan. The results of patch/release tests shall be recorded. The PMT analyzes and corrects discrepancies found in build testing. It is recommended that the test team produce a test report after the completion of testing for the patch/release.

2.5.1.6 Prepare Final Updates to System and User Documentation

The SMP identifies user and system documentation that is to be maintained. At the conclusion of release testing these documents are updated according to project standards and placed under the specified level of configuration control.

2.5.2 Acceptance Testing

2.5.2.1 Conduct a Review of the Test Results

It is recommended that an informal ATRR be held for each release. The test team produces a test report after the completion of testing for the release and reviews the results with customer and user representatives. In most cases a PMT-level review of the release/patch test results is either informal or optional. For large PMT's or critical changes a formal review is recommended. Minutes of the meeting shall be taken and RFA's tracked to closure.

If any external individuals (not Civil Servants) participate in the ATRR, appropriate non-disclosure agreements must be signed in advance of the review.

ATRR:

- List of maintenance requests in the release, including priority and test status (pass/fail)
- Summary of tests conducted and tests passed or failed.
- List of nonconformances remaining in the release and operational impact of the nonconformances on operations.

2.5.2.2 Support Acceptance Testing Performed by the Customer or by an Independent Organization

It is recommended that members of the PMT be available to support Acceptance Testing as conducted by the customer or an independent test organization. This support consists of consulting activities and analysis of test results as requested.

2.6 Delivery and Support Activities

2.6.1 Delivery

For external releases or deliveries, additional requirements may apply. See NPD/NPR 2210.1, “External Release of NASA Software,” for further information on the domestic or foreign release of software created by or for NASA.

2.6.1.1 Prepare the Release Letter

The PML shall document the delivery of the accepted system release in a release letter to the customer that details the release identification, release contents, release capabilities, and any remaining nonconformances. The release letter identifies any changes that affect operations or require user training. All software and documentation that are part of the delivery are promoted to the configuration-controlled delivery library at the time of delivery and constitute a system baseline.

2.6.1.2 Prepare Shipping Records for Delivery

The PMT shall prepare shipping records that accompany any items transported. The records identify each item being delivered, its delivery medium, and its destination.

2.6.1.3 Deliver the Completed System or Release

The PMT shall deliver the software and all documentation using the media and methods identified in the Software Management Plan. Release documentation includes installation and back-out instructions. (See NPD/NPR 2210.1 regarding external releases or deliveries.)

2.6.2 Operations Support

2.6.2.1 Support Review of the Acceptance Test Results

It is recommended that the PMT support the ORR if one is held. The ORR serves as an opportunity to review the ability of the system to support operations. The status of remaining nonconformances is presented.

ORR Topics:

- Summary of Acceptance Testing including nonconformances and their impact on operations
- Summary of tests conducted, tests passed and tests failed
- Identification of nonconformances and suitability of workarounds in the operational environment

DIRECTIVE NO. GPR 8700.5A
EFFECTIVE DATE: January 5, 2005
EXPIRATION DATE: July 19, 2009

Page 36 of 51

2.6.2.2 Provide User Training

To the extent specified in the SMP, the PMT shall provide user training and installation instructions, and user documentation specific to new or modified user and operations capabilities.

2.6.2.3 Provide Technical Assistance

The PMT shall provide technical support to the extent specified in the SMP. The PMT responds to questions raised by operations staff and other users and assists in identifying and analyzing defects or anomalies in the behavior or performance of the system.

Attachment - Software Management Plan Guide

The following sections detail activities required by the QMS for the **in-house** development or maintenance of software products that are within scope of the QMS. Although the activities listed here should always be performed, the PDT or PML should tailor the extent and degree of the activities performed based on product complexity and criticality. Although the steps are listed sequentially, some may be actually performed in parallel, or even on an iterative basis. As part of the development process, a Software Management Plan shall be developed and maintained under Project or line management control. The information described below has been structured to allow for use directly in a single SMP document. The information may be in separate documents or may be included in another document at the discretion of the PDL or PML. However, all information indicated as required shall be readily available.

For administrative or institutional software, the responsible Director of may grant waivers to these requirements, using the process described in GPR 1280.1. Note: Any software used in any flight or ground system is not considered institutional.

Records indicated below are kept by the PDL or Project as determined by the Project Manager or by other GPR's.

Software Management Plan Contents

The items below are required (i.e., are to be interpreted as “shall”) unless otherwise noted.

It is recommended that items identified with an “*” be included by reference and not detailed in the SMP.

Controlled Document	Comment	Maintained By
Software Management Plan	Signed and dated by the Configuration Control Board (CCB) that is listed in Plan—usually PDL, Branch Head, and Customer.	PDL or Project

1.0 Customer Commitment and Requirements

Section 1 is to include only those items specified by the customer, not information derived by the PDT.

1.1 Background (Recommended) -- Include a brief description of what larger effort/activity this PDT is supporting and how this product fits into the larger picture.

1.2 PDT Charter (Recommended) -- Include a brief description of what this PDT is being asked to accomplish, including any time constraints or interface boundaries within which this PDT is expected to operate.

1.3 Customer(s) Identification (Recommended) -- The customer is generally the person who pays the bill. It could also be the person who will define the requirements and accept the products.

CHECK THE GSFC DIRECTIVES MANAGEMENT SYSTEM AT
<http://gdms.gsfc.nasa.gov> TO VERIFY THAT THIS IS THE CORRECT VERSION PRIOR TO USE.

1.4 Customer Goals and Objectives (Recommended) -- List any special things that the customer wants to accomplish (e.g. rapid turn around, new architecture, special COTS requirements, special experiments, etc.) through this Team's activities.

1.5 Customer Requirements* -- This section should reference (preferred) or list the high-level requirements *specified by the customer*, and should include any *customer-specified* standards to be met or interface control documents needed. Do not include derived technical interface documents or databases here; reference them in Section 3.1.1.

Controlled Document	Comment	Maintained By
High-level Requirements	Signed and dated by Customer	Project or PDL

1.6 Deliverables -- List *customer-specified* products to be delivered for each phase of development, including software, hardware, licenses, documentation, etc.

1.7 Schedules* -- List *customer-specified* schedule requirements, including such items as documentation, releases and reviews.

1.8 Necessary Customer Training (Recommended) -- Specify who is to be trained, how many are to be trained, location and nature of training.

1.9 Medium/Method for Product Delivery (GPR 6400.1) (Recommended) -- List any *customer-specified* delivery medium and method of delivery for all products listed in Section 1.6. Describe those media/methods that are NOT specified by the customer in Section 3.2.

1.10 Product Destination (Recommended) -- List product delivery destinations for all products listed in Section 1.6.

1.11 Post- Delivery Maintenance -- Describe maintenance requirements as specified by the customer.

1.12 Customer-supplied Elements, Both Technical and Resource-related (schedule, medium, and interfaces) -- List any technical elements supplied by the customer that will be used in the production, testing, packaging or delivery of the product. Do not include funding. Include delivery schedule and medium of supplied items.

1.13 Customer Involvement (roles, responsibilities, authority, accountability) (Recommended) -- Provide details on the extent of direct customer involvement with the PDT (e.g., "Attends PDT meetings," "Reviews results," "Provides direction,").

1.14 Acceptance Criteria* -- Describe the customer's criteria for determining when the product is completed (e.g., "When will the customer accept the product?"). This is usually demonstrated by having a satisfactorily completed test matrix/set of test plans. The customer's verbal acceptance is not sufficient.

1.15 Customer Requirements Review and Update Process – Describe the process used to evaluate and approve changes to the customer requirements. Be sure to note that the PDT will be evaluating the changes to assure that they have the capability of providing the requested changes within the allotted resources and schedule. Approval authorities (those listed on the signature page) must be specified by name and title. It must be stated whether or not the approval authorities consist of the CCB. If the approval authority is the CCB, then the CCB process and membership must be described or referenced. If the approval authority is not the CCB, then specify the approval authority and describe or reference its process and membership. The original approval authority must also approve any changes.

2.0 Development Planning (GPR 8700.1)

2.1 General Development Approach (Recommended) -- Describe briefly the general philosophy that will be used to build the product, discussing such aspects as use of COTS, contractor involvement, schedule constraints, use of a particular development methodology or new technology, etc.

2.2 Resources Needed (budget, people/skills, and facilities)* -- Indicate where the *official* budget is kept. In most cases, the budget will probably reside with the Project. Budget information should be kept by fiscal year, and shall include both civil servant staffing and any contractor support. Address any specific facilities or any facility modifications required for use in development or testing, and the dates they are expected to be required. .

Working Documents	Comment	Maintained By
Budget	May be included in SMP by reference.	PDL or Project

2.3 PDT Information

2.3.1 PDT Organization* -- Include an explanation or diagram illustrating the organization of the PDT personnel and its activities. Include the relationship of the PDL to the higher-level Project organization for status and accountability, if applicable.

Working Documents	Comment	Maintained By
PDT organization	May be included in SMP by reference.	PDL or Project

2.3.2 Roles, Responsibilities, Authority, Accountability of PDT Members* -- Describe the process used to assign work to PDT members and document the work assignments. Assignments may be made by subsystem (e.g., Command & Data Handling, Planning & Scheduling) or by work function (e.g., testing).

Working Documents	Comment	Maintained By
PDT work assignments	May be included in SMP by reference.	PDL

2.3.3 PDT Interfaces to Other Teams, Organizations, or Groups (Recommended) -- Describe any interfaces to other organizations, teams, or groups necessary in developing the product, and provide a brief description of the purpose of each interface. This may include activities such as the interface of the

flight software group to the flight hardware group for working compatibility issues, or the interface of the Ground Data System to the Flight Operations Team for acceptance of the system.

2.3.4 PDT QMS Training Plan* (GPR 3410.2) -- Identify any QMS required task-specific training need for each PDT member. When training is complete, document it by keeping a list of name, course, and date completed. (QMS required task-specific training is defined as training that must be taken to acquire new skills or enhance current skills required to perform tasks of that position that affect quality. This includes familiarizing PDT members with the Software Management Plan, the methodology, standards, and design process used, the team records, and use of a nonconformance recording system or training required when working in proximity to instruments or spacecraft, such as Electrostatic Discharge Awareness Training, Range or Launch Safety, Laser Safety, etc. It does not include “developmental” training (e.g., in a software language or tool).

Record Title	Record Custodian	Retention
Records of required training needed	Supervisor/PDL/Product Manager	NRRS Schedule 3/19 - Records not authorized for disposal at this time
Records of required training completed	Supervisor/PDL/Product Manager	NRRS Schedule 3/19 - Records not authorized for disposal at this time

2.4 Procurement* (GPR 5100.1) -- Describe all hardware and software purchase requirements in detail (e.g., the specifications for a product to be acquired). Include any purchases necessary for facility modification. If you are using contractor support, list the contractor name and contract number. If special or unusual contracting arrangements are required, describe them. Reference the procurement process.

Working Documents	Comment	Maintained By
Purchase requests		PDL or PDT procurement person

2.5 Risk mitigation* (Continuous Risk Management) -- Describe any areas where there is a special risk to the delivery of the product and describe how these risks will be continuously identified, analyzed planned for, and tracked. A detailed discussion of the NASA process and resources for Continuous Risk Management is found in NPR 7120.5, “NASA Program and Project Management Processes and Requirements.” If risks are described in a separate document or within the larger Project Risk Mitigation Document, reference the document here. If there are no risks, state that.

2.6 Security (NPR 2810.1) -- Describe the plans for addressing security considerations, both physically for the facilities involved and electronically for any computer systems being used either for development and testing or as a part of the final product. Provide inputs to the Project Security Plan, as needed.

2.7 Independent Verification and Validation (IV&V) -- The PDL shall provide the appropriate documentation and software information to the Project Manager to assess the IV&V requirements for the mission. Once the IV&V requirements have been determined and documented by the Project, the PDL is responsible for interfacing with the IV&V facility and providing any required support to the IV&V

activity. Describe the plan, or reference the Project level plan, for providing any required IV&V support.

2.8 Review Program (GPR 8700.4/GPR 8700.6) -- Describe the types of reviews you plan to have and the membership of the review boards. GPR 8700.4 and GPR 8700.6 define the procedures and guidelines for required mission reviews and their applicability. The software PDT participates in, or contributes material to, required mission-level Project or Program reviews (e.g. EPR's or IIR's), as needed. The PDL shall define, with the participation of the Product Manager, line management and the customer, an appropriate set of software reviews as a resource to increase the probability of success. Reviews may be combined to improve value or efficiency. However, when reviews are combined, review objectives from each shall be addressed to the level of detail required for the individual reviews. In addition include a discussion of any code or design walkthroughs you plan to use.

Working Documents	Comment	Maintained By
Review plan	May be included in SMP	PDL or Project

Record Title	Record Custodian	Retention
RFA's and responses from review meetings	PDL or Project Manager	NRRS 7/5B1 - Permanent. Document may be retired to FRC 1 year after publication. Transfer to NARA when 25 years old.
Review meeting notes with action item list and resolutions	PDL or Project Manager	NRRS 7/5B1

2.9 Overall Schedule* -- This should be the overall schedule used to manage PDT activities. It should contain the PDT lifecycle schedule including facility preparations, procurements, system development by phase and build/release, product delivery, and maintenance (if applicable). Be sure to include schedule dates for reviews, documentation, delivery of interface control documents (ICD's), tests, software releases, procurements, and external deliveries to the customer. The detailed schedule should include and be consistent with any customer-specified schedules defined in Section 1.7.

Working Documents	Comment	Maintained By
Schedules	May be included in SMP by reference.	PDL or Project

3.0 Technical Approach

3.1 Product Development (GPR 8700.2)

3.1.1 Derived Requirements* -- Reference (preferred) or describe the requirements and/or specifications derived from customer requirements by the PDT and approved by the Customer, PDL and appropriate manager. These should include assumptions, interfaces, and performance information. Ensure that requirements are testable. If no additional requirements/specifications can be derived, just reference Section 1.5.

Controlled Document	Comment	Maintained By
Derived requirements	Signed and dated by Customer, PDL, and appropriate manager	Project or PDL
Interface control documents	Signed and dated by representatives of interfacing organizations	Project or PDL

3.1.2 Development Strategy

3.1.2.1 Development Process* -- Reference (preferred) or describe at a high-level the tailored lifecycle and development process you will use. Tailor the standard GSFC Process for Software Development or Process for Software Maintenance in the body of this GPR to create your process. Tailoring is determined by the PDL with the concurrence of line management and the customer (typically a project manager or principal investigator). Tailoring factors include project characteristics such as the criticality of the application, the size of the Product Development Team (PDT) and user community, the degree of reuse, and other project specific factors. Required activities are designated as "shall" and may not be deleted. Best practices from past experience are designated as "recommended" and should be thoughtfully considered for adoption by the PDL.

Working Documents	Comment	Maintained By
Tailored development process		PDL or Project

3.1.2.2 Development Environment -- Describe the development and test hardware and locations, and all PDT development standards, as appropriate. Describe plans (such as back-ups) to prevent loss or damage to all of the products (including software, documentation and hardware) during all phases of development.

3.1.2.3 Buy Approach* (GPR 5100.1) -- Describe any special purchasing strategies for items specified in Section 2.4. This may include strategies for use of COTS such as agreements for vendor modifications to address specific requirements. (Note that in most cases, open competitive procurement of hardware and software products is required. See GPR 5100.1 for additional information on procurements.)

3.1.2.4 Customer Supplied Products Approach -- Briefly describe the approach that will be used to integrate any customer-supplied items (as specified in Section 1.12) required for the development and test of final product. Identify any assumptions concerning these items or their integration.

3.1.2.5 Prototyping Approach* -- Describe any prototyping activities required to develop the product and the purpose of the prototype (i.e., "What specific questions are to be answered by the prototype?")

3.1.3 Product Design* -- Reference (preferred) or briefly describe the design of the product that the PDT is planning to produce. Describe how changes in design are incorporated and traced to changes in

the requirements. Describe how the design will be traced to the customer-specified and derived requirements/specifications.

Record Title	Record Custodian	Retention
Documentation of design verification activities Include one or more: <ul style="list-style-type: none"> • Design baseline • Design walkthrough documentation (including issues and resolutions) 	PDL or Project	NRRS 8/5A2 - Records may be retired to a FRC when 2 years old. Destroy when 15 years old.
Completed design documentation May include: <ul style="list-style-type: none"> • High-level architecture description • Design review materials • Design documents 	• PDL or Project	NRRS 8/5A2 - Records may be retired to a FRC when 2 years old. Destroy when 15 years old.

3.1.4 Build Approach* -- Include the development phases, the sequence of builds, vendor/customer/prototype elements to be integrated, and the requirements satisfied in each build.

Record Title	Record Custodian	Retention
Completed build plan	PDL or Project	NRRS 8/5A2 - Records may be retired to a FRC when 2 years old. Destroy when 15 years old.

3.1.5 Product Testing

3.1.5.1 Incoming Inspection and Test (GPR 4520.2) -- For purchased items, including hardware and flight-critical components, document the Receiving Inspection Instructions to describe special receiving instructions and tests other than kind, count and condition.

Record Title	Record Custodian	Retention
Receiving inspection instruction	As per GPR 5330.1	NRRS 5/10 - My be destroyed when 2 years old

3.1.5.2 Product Inspection and Test * (GPR 5330.1) -- Describe the testing approach from unit testing through product delivery (including in-process and final inspection). Reference your test plans and discuss your testing approach for unit, build/release, and system testing; the composition of the test team (developers, independent, customer); test data (simulator, supplied data, flight hardware, real data); and any related success criteria (particularly any from the customer for final acceptance—see Section 1.14). Describe how changes in requirements and design are mapped to changes in test plans.

Controlled Documents	Comment	Maintained By
Documentation of planned verification activities	Include one or more: <ul style="list-style-type: none"> • Unit tests • Build/release test plans 	PDL or Project
Documentation of planned (end-to-end) system validation activities	Include one or more: <ul style="list-style-type: none"> • High-level description of test to be run (can be in test procedures) • Test validation matrices • Acceptance criteria 	PDL or Project

Record Title	Record Custodian	Retention
Documentation of verification activities Include one or more: <ul style="list-style-type: none"> • Unit test results • Code reading actions and dispositions Build/release test results	PDL or Project	NRRS 8/5A2 - Records may be retired to a FRC when 2 years old. Destroy when 15 years old.
Documentation of (end-to-end) system validation activities Include one or more: <ul style="list-style-type: none"> • Validation matrices • Validation test results 	PDL or Project	NRRS 8/5A2

3.1.5.3 Statistical Techniques* (GPR 8070.2) -- Unless the PDT determines a need for statistical testing of the product or other statistical methods, include the following paragraph in this section of your SMP.

“The PDT has evaluated the need for statistical testing of the products developed under this Software Management Plan and has determined that statistical techniques are not required.”

Examples of statistical techniques being used are (1) techniques to obtain reliability of hardware systems and (2) comparisons of output results after a platform language conversion. If statistical techniques are being used, then the procedure for their use must be documented.

3.2 Product Delivery -- List any medium/method specified in Section 1.9 by which the various products are to be delivered to the customer. State that products released to the customer will include a *release letter* listing the release number, the included capabilities of the release, and a description of any remaining nonconformances in the release.

Record Title	Record Custodian	Retention
Product release letters	PDL or Project	NRRS 8/5A2 - Records may be retired to a FRC when 2 years old. Destroy when 15 years old.
Shipping records	PDL or Project	NRRS 8/5A2

3.3 Servicing -- Describe the process for post-delivery product maintenance, including processes for maintenance specified by the customer in Section 1.11 and any derived maintenance requirements (i.e., "How do you plan to meet the requirements specified in Section 1.11?"). Address responsibility for the maintenance. Describe the processes that will be used for handling maintenance requests, for doing work, and for product redelivery of custom, GOTS and COTS software, and hardware.

Record Title	Record Custodian	Retention
Maintenance requests	PDL	NRRS 7/21B2 - Destroy when no longer needed or when 2 years old, whichever is later.
Redelivery letters	PDL or Project	NRRS 7/21B2

4.0 Management Approach

4.1 Status Tracking

4.1.2 Design/Implementation Status -- Describe the method(s) that will be used to track status through design and implementation of the product.

Working Documents	Comment	Maintained By
Status information	May include: <ul style="list-style-type: none"> • Schedule charts with status indicated • Module-by-module checklist • Configuration management records • Documentation of regular status meetings 	PDL or designee or Project

4.1.3 Testing Status -- Describe the method used to track testing status of the product throughout its life cycle.

Working Documents	Comment	Maintained By
Test status information	May include: <ul style="list-style-type: none"> • Test status chart • Documentation of regular test status meetings • Signoffs of completed tests in test plan or procedures 	PDL or designee or Project

Metrics Collection and Analysis* -- The project shall collect the following metrics: schedule dates, budget (effort and cost), product size, and product error information such as the number of open/closed nonconformances (i.e., discrepancy reports). PDT process metrics, Project/Organizational metrics, and additional product metrics may also be collected. Describe how you will use these metrics for process improvement. NASA metrics shall also be collected in accordance with existing guidelines.

Record Title	Record Custodian	Retention
Data and completed forms representing the required metrics	PDL or Project	NRRS 7/21B2 - Destroy when no longer needed or when 2 years old, whichever is later.

4.2 PDT Lessons Learned -- The PDL shall query the NASA Lessons Learned Information System (which is maintained at <http://llis.nasa.gov>) and other knowledge resources, as appropriate, to access relevant past experiences and knowledge that can be leveraged to reduce risk, improve quality and efficiency. These queries shall be conducted at the beginning of and then periodically throughout the software development lifecycle. The PDL shall submit significant lessons learned to the web-based LLIS throughout the product lifecycle, as appropriate.

Describe the process to be followed to ensure capture by the PDT of relevant lessons learned throughout the entire life cycle. The final lessons learned report is intended to be a brief summary of the PDT's key recommendations for improving the development/maintenance process in future similar projects.

Working Documents	Comment	Maintained By
Lessons-learned log or document		PDL or Project

4.3 Key Issues, Decisions, and Rationale (Recommended) -- Maintain a log of the PDT's key issues, decisions, and rationale throughout the life cycle.

Working Documents	Comment	Maintained By
Log of key issues, decisions, rationale		PDL

5.0 Product Assurance

5.1 Control of Nonconforming Products and Corrective Action* (GPR 5340.2/GPR 1710.1) -- Describe the criteria and process for recording and correcting problems in a “minor” NCR/CA system. Include a description of the process used to evaluate the cause of the problem and to assess whether any changes need to be implemented to prevent future recurrences. The minor NCR system should include the version or release number where the problem was found and, ideally, the version number that includes the corrections. Any nonconforming products released to the customer shall be identified as such in the release letter (Section 3.2), which shall describe the remaining nonconformance. The release number and the list of NCR's associated with the release identify Nonconforming products. Products with remaining nonconformance may only be released to the customer with proper approval. (After product delivery the Center NCR/CA system shall be used if no minor nonconformance system exists or if the nonconformance meets the criteria for a major nonconformance as specified in GPR 1710.1 for use of the Center NCR/CA system.)

Record Title	Record Custodian	Retention
Nonconformance records from minor NCR system (Records not in NCRCAS)	PDL or Nonconformance Lead	NRRS 5/30B - Close file at end of FY. Destroy when 9 years old.
Nonconformance records from Center NCR system	Code 302, Systems Safety and Reliability Office (NCRCAS)	NRRS 5/30B
Corrective action plans - May be in NCRCAS	PDL or Nonconformance Lead	NRRS 5/30B

5.2 Control of PDT Software, Hardware, Documentation, and Data*(GPR 1410.2) -- Describe how your PDT does configuration management for your software, hardware and documentation and who has the change authority for each, including identification of any CCB's and the items controlled. If you use the Project’s process for any of those, reference where their procedures can be found. Describe the signature and change authority for the SMP. Describe the method used to uniquely identify versions of the software and the elements from which it is built. The use of a commercial configuration management tool is strongly recommended for environments where one is available. If on-line copies of documentation or software are considered the controlled copy, then the approval authority should control on-line change access.

In the SMP identify the types of documents that will be controlled and when each type will be placed under configuration control. A list of documents and data under configuration management by the PDT is to be referenced in the SMP in this section. The list is to include the document or database name, the date or version identification of the current version, the location of the documents or database, and the person responsible for the item. NOTE: Any databases or web sites containing controlled information directly under the PDT’s control should contain a header identifying what is being viewed, as well as the date of the last change and the person responsible for its control.

Record Title	Record Custodian	Retention
Software CM records	PDL (or Configuration Manager) or Project	NRRS 8/5-A2 - Records may be retired to an FRC when 2 years old. Destroy when 15 years old.
List of items under configuration management	PDL (or Configuration Manager) or Project	NRRS 8/5-A2
Copy of signature page of configuration management item	PDL (or Configuration Manager) or Project	NRRS 8/5-A2
Records of CCB approval	PDL (or Configuration Manager) or Project	NRRS 8/5-A2

5.3 Control of Test Software and Hardware (GPR 8730.1) -- Describe anything (hardware and/or software) used to test the product. Describe how test software will be validated (i.e., how do you convince yourself that the simulator is working properly?). If the software used for testing is not the final validation, but is only used as part of a self-check, where neither the test software or the product being tested is considered correct until the final results are correct, then describe that test capability and any associated limitations. Describe or reference the configuration management process used to ensure the appropriate version of the test hardware/software is used. Also discuss any inspection, measuring, and test equipment (IMTE) being used and any calibration requirements.

Working Documents	Comment	Maintained By
Documentation of planned test software verification activities	Include one or more: <ul style="list-style-type: none"> • Test plans • Acceptance criteria 	PDL or Project

Record Title	Record Custodian	Retention
Calibration records and calibration due dates for any IMTE used by the PDT	As per GPR 8730.1	NRRS 8/5-A2 - Records may be retired to an FRC when 2 years old. Destroy when 15 years old.
Test software test results	As per GPR 8730.1	NRRS 8/5-A2
Records of verification from contractor	PDL or Project	NRRS 8/5-A2

5.4 Control of Customer Supplied Products* -- Describe the method that will be used to check out or test the software or hardware supplied to you by the customer for inclusion into the product or for testing or packaging of the product. If a customer provides items for use by the PDT in the development/testing of the product, describe the process used to report any problems with the item(s) back to the customer. Describe the configuration management process for changes (initiated by the customer or the PDT) to customer-supplied elements listed in Section 1.12. Include any other processes used to safeguard customer-supplied products. This section should address simulators, test data, software algorithms, software and/or hardware received from the customer.

DIRECTIVE NO. GPR 8700.5A
EFFECTIVE DATE: January 5, 2005
EXPIRATION DATE: July 19, 2009

Record Title	Record Custodian	Retention
Problem reports on customer-supplied products	PDL or Project	NRRS 7/21B2 - Destroy when no longer needed or when 2 years old, whichever is later.

5.5 Records* (GPR 1440.7) -- Identify the PDT's records to be maintained (defined in GPR 1440.7) and the person who will be responsible for them.

ACRONYMS

ARM	Automated Requirements Measurement
ATTR	Acceptance Test Readiness Review
CCB	Configuration Control Board
CDR	Critical Design Review
CM	Configuration Management
COTS	Commercial off the Shelf
EPR	Engineering Peer Review
GOTS	Government off-the-Shelf
GPR	Goddard Procedural Requirement
GSFC	Goddard Space Flight Center
HCI	Human Computer Interface
ICD	Interface Control Document
IIR	Integrated Independent Review
IMTE	Inspection, Measuring and Test Equipment
ISO	International Standards Organization
IV&V	Independent Verification and Validation
LLIS	Lessons Learned Information System
NASA	National Aeronautics and Space Administration
NCR/CA	Nonconformance Reporting/Corrective Action
NPD	NASA Policy Directive
NPR	NASA Procedural Requirements
ORR	Operational Readiness Review
PDL	Product Development Lead
PDR	Preliminary Design Review
PDT	Product Development Team
PM	Product Manager
PML	Product Maintenance Lead
PMT	Product Maintenance Team

DIRECTIVE NO. GPR 8700.5A
EFFECTIVE DATE: January 5, 2005
EXPIRATION DATE: July 19, 2009

Page 50 of 51

QMS	Quality Management System
RFA	Request for action
SCR	System Concept Review
SMP	Software Management Plan
SRR	System Requirements Review
SSR	Software Specifications Review
TBD	To be determined

DIRECTIVE NO. GPR 8700.5A
EFFECTIVE DATE: January 5, 2005
EXPIRATION DATE: July 19, 2009

CHANGE HISTORY LOG

Revision	Effective Date	Description of Changes
Baseline	07/19/04	Initial Release
A	01/05/05	As directed during the FY04 Center Rules Review, the Responsible Office modified this document to remove requirements that were no longer needed and to clearly distinguish requirements from supporting information. Administrative changes were made throughout to correct responsible organization names and codes, and to retitle Goddard Procedures and Guidelines (GPG) to Goddard Procedural Requirements (GPR). All changes were reviewed and approved by the Goddard Quality Management System Council (QMSC).